Neural Network and Deep Learning

Ariful Islam

July 2025

1 Introduction

This report is about five task about autoencoder in Deep Learning.

2 Questions

- Train an autoencoder as a 2D feature generator and display the features extracted from the CIFAR-10 dataset on a 2D plane.
- Compare the autoencoder-generated features with features extracted by a pre-trained CNN. Use dimensionality reduction techniques such as:
  - Principal Component Analysis (PCA)
  - t-distributed Stochastic Neighbor Embedding (t-SNE)
- Train a denoising autoencoder on the CIFAR-10 dataset by adding noise to the input images and learning to reconstruct the original clean images.
- Train a CNN-based classifier on the CIFAR-10 dataset **without** applying any single-image data augmentation techniques.
- Train a CNN-based classifier on the CIFAR-10 dataset **with** single or multiple data augmentation techniques applied to individual images (e.g., rotation, flipping, cropping).

3 Answer:

3.1 Task 1:

Here, we trained an autoencoder on CIFAR-10 to learn compressed 2D features from images. We added a 2D latent layer and visualized the features to see how well the classes were separated.

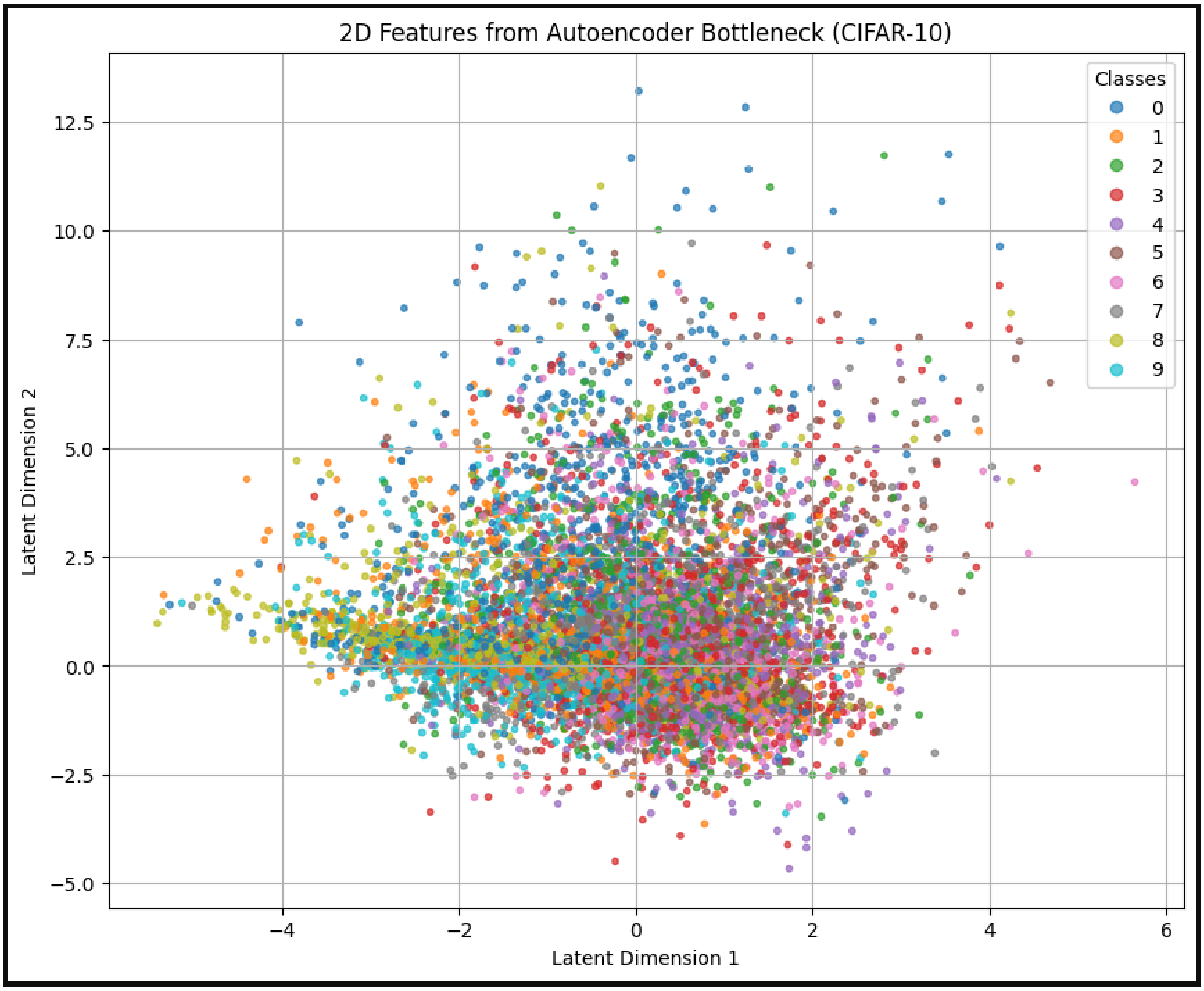


Figure 1: 2D features from autoencoder

3.2 Task 2:

In this task, we used a pretrained CNN to extract features from CIFAR-10 images. Then, we applied PCA and t-SNE to reduce the feature dimensions and visualize how the data clusters.

We extracted CIFAR-10 features from a pretrained CNN in two ways: without training and by fine-tuning some layers for 10 epochs. Then, we used PCA and t-SNE to reduce and visualize the feature dimensions for both cases.

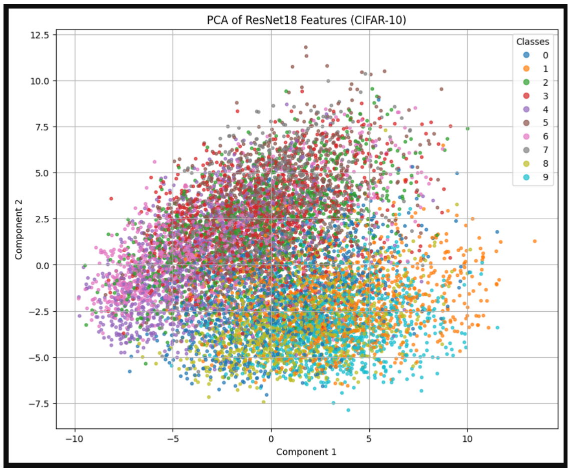


Figure 2: *
Image 2

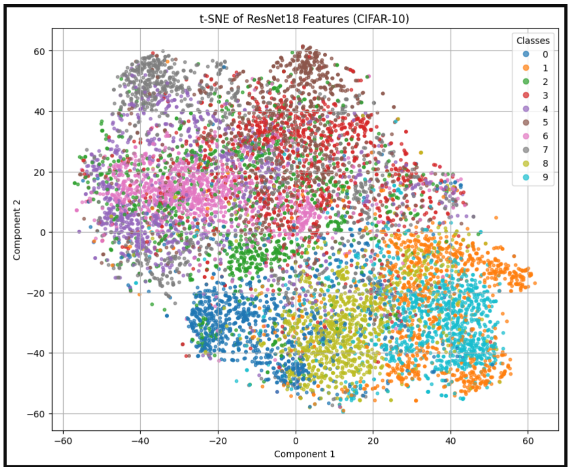


Figure 3: *
Image 3

Figure 4: Features of CIFAR-1O in ResNet18 without tuning

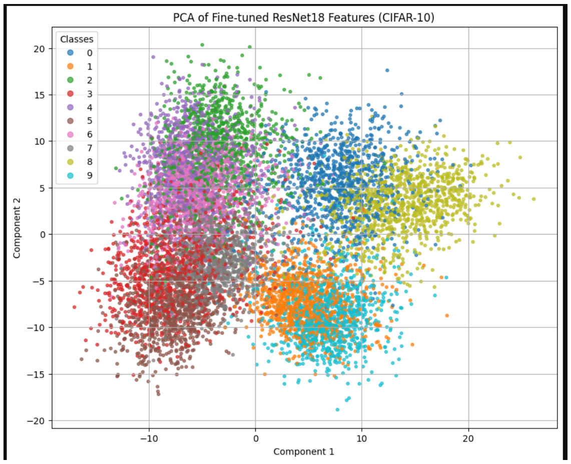


Figure 5: *
Image 2

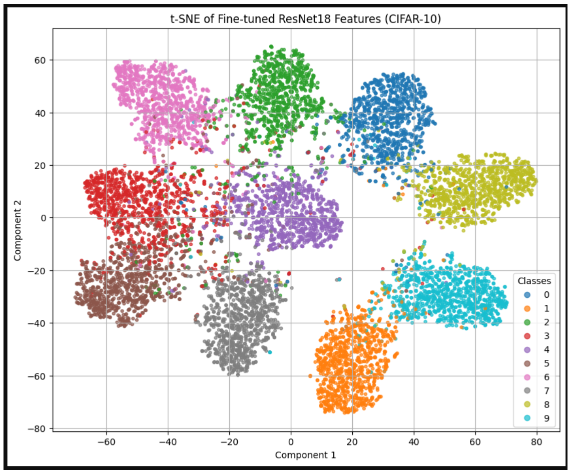


Figure 6: *
Image 3

Figure 7: Features of CIFAR-1O in ResNet18 with tuning

3.3 Task 3:

In Task 3, we trained a denoising autoencoder to remove noise from corrupted CIFAR-10 images. We added noise to the inputs and then visualized how well the model could reconstruct the clean images.

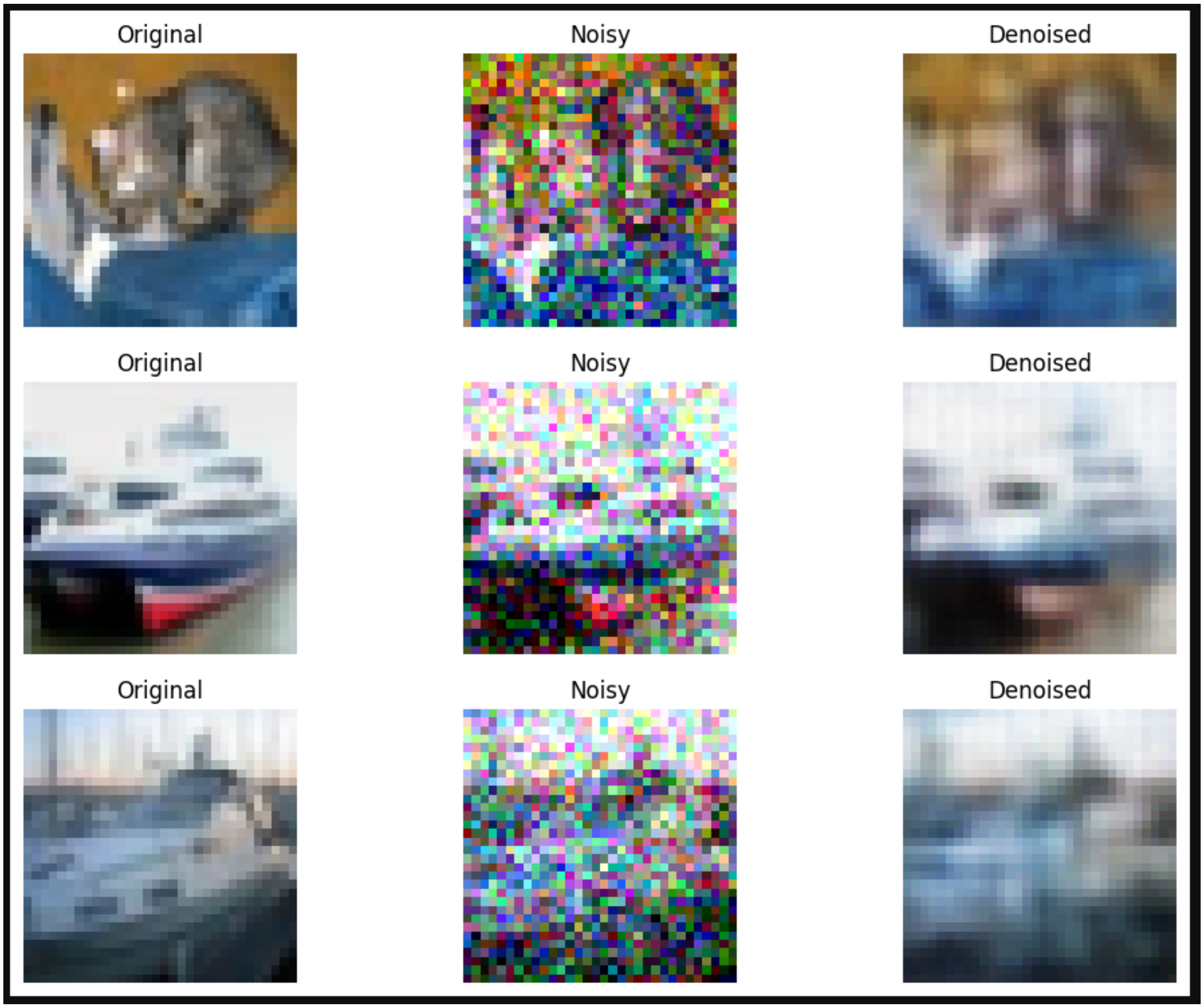


Figure 8: The figure shows the Origina image Noisy image and denoised output from denoised-encoder

3.4 Task 4:

Here, we trained a CNN classifier on CIFAR-10 without applying any data augmentation techniques. This helped us understand the baseline performance of the model on raw images.

The model achieved a high test accuracy of 94.79 percent, showing strong generalization on CIFAR-10. Training loss steadily decreased while accuracy rapidly improved, reaching nearly 100 percent training accuracy by the 10th epoch.

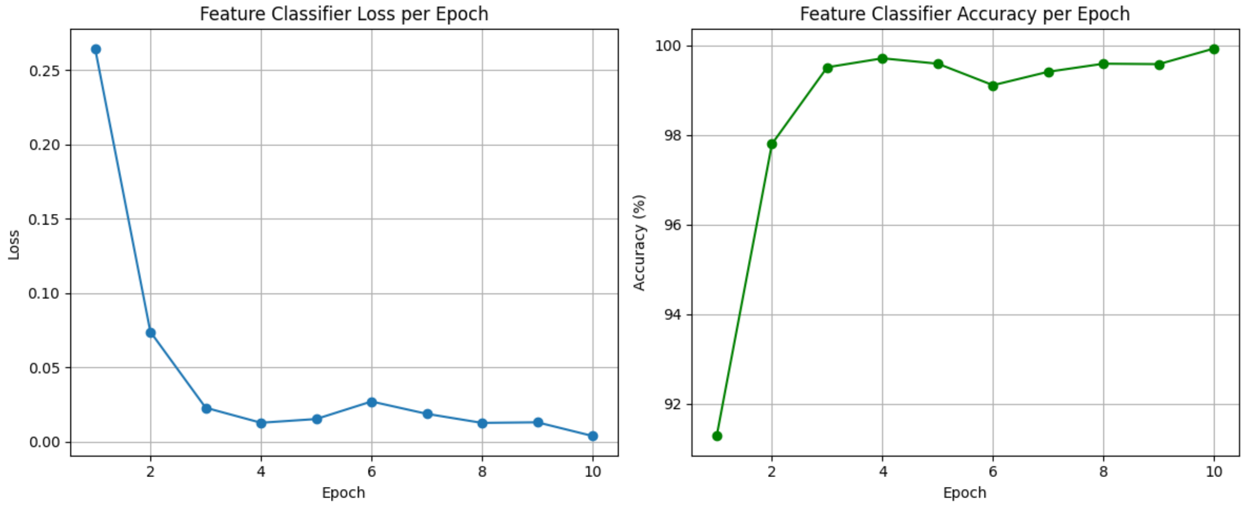
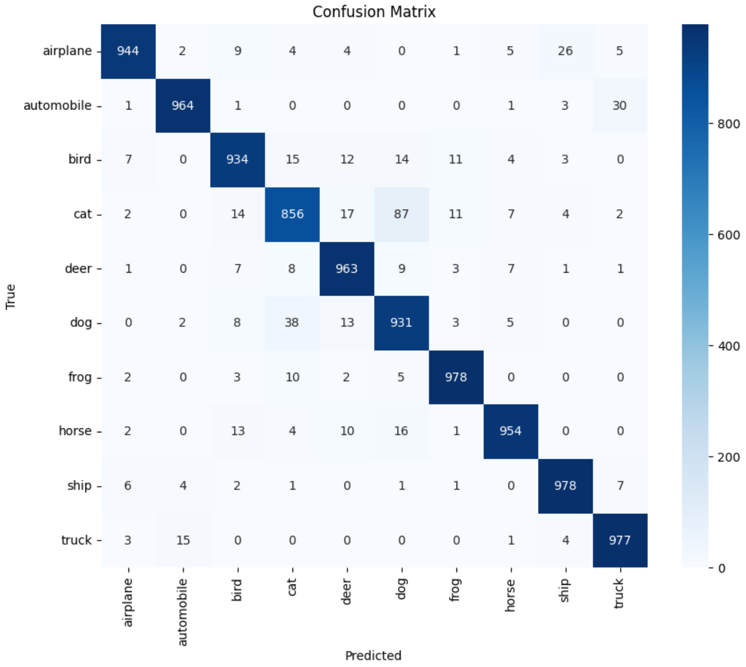


Figure 9: The figure shows Training loss vs Training accuracy.



	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	944	2	9	4	4	0	1	5	26	5
automobile	1	964	1	0	0	0	0	1	3	30
bird	7	0	934	15	12	14	11	4	3	0
cat	2	0	14	856	17	87	11	7	4	2
deer	1	0	7	8	963	9	3	7	1	1
dog	0	2	8	38	13	931	3	5	0	0
frog	2	0	3	10	2	5	978	0	0	0
horse	2	0	13	4	10	16	1	954	0	0
ship	6	4	2	1	0	1	1	0	978	7
truck	3	15	0	0	0	0	0	1	4	977

Figure 10: The figure shows the confusion matrix of the CNN.

3.5 Task 5:

In Task 5, we fine-tuned a pre-trained ResNet-18 on CIFAR-10 using data augmentation techniques such as random crop, flip, and rotation. The pretrained ResNet18 fine-tuned with data augmentation achieved a test accuracy of 67.31 percent, Using augmentation helped reduce overfitting by exposing the model to varied data, leading to more stable learning and better generalization.

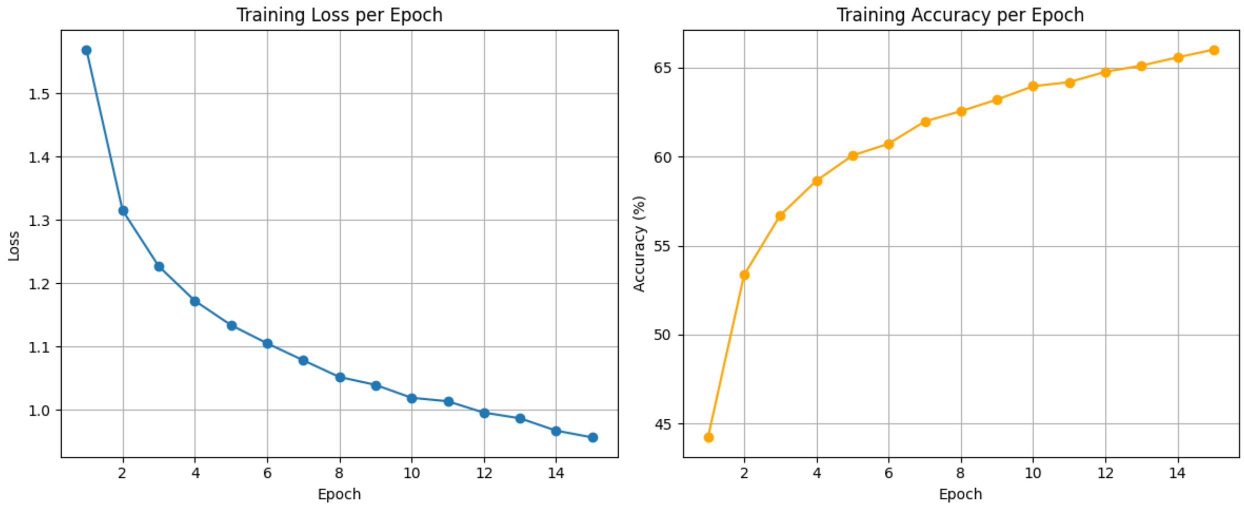


Figure 11: The figure shows Training loss vs Training accuracy.

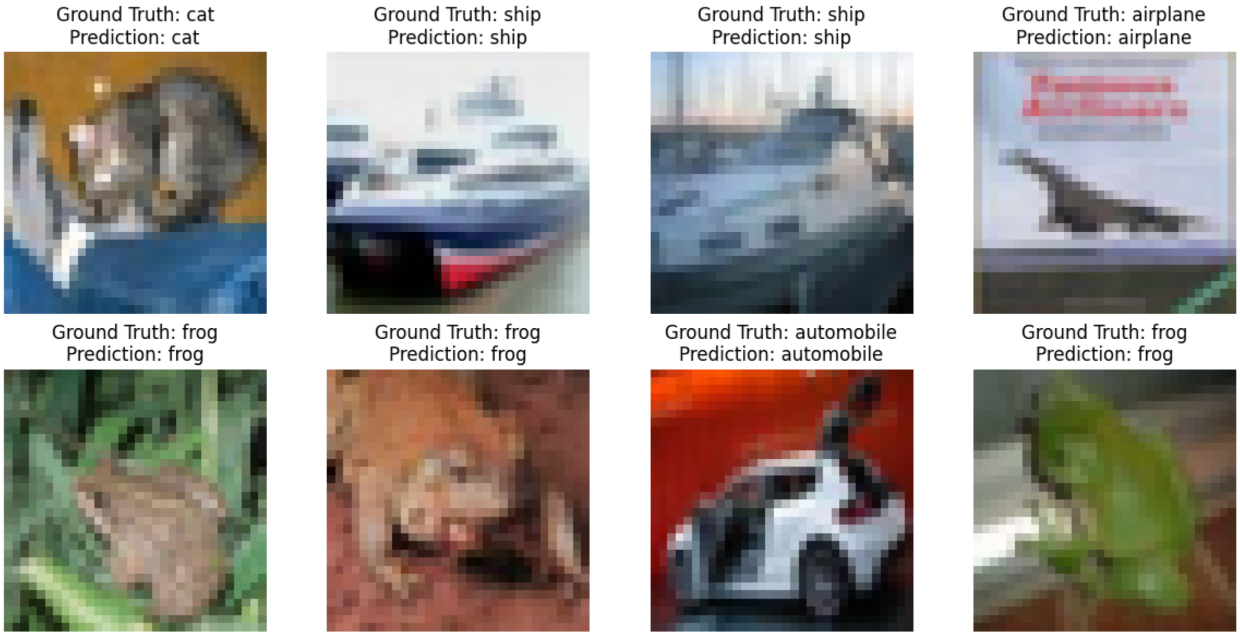


Figure 12: The figure shows some images, Ground truth, and prediction value.

4 Source Code Link

GitHUb Link-https://github.com/Arif111866/Deep-Learning-AI